REMARKS

Claims 1-24 were pending. Claims 1-11 are allowed while claims 12-24 are rejected. The Applicant has amended claims 12, 13 and 15 and cancelled claim 14. Therefore, claims 1-13 and 15-24 are presently pending. The Applicant requests further consideration and re-examination in view of the amendments above and remarks set forth below.

Claim 12 is amended as explained below. Claims 13 and 15 are amended and claim 14 is cancelled to comport with the amendments to claim 12.

Amendment to the Specification

The Examiner requested the Applicant to indicate that Figures 2 and 3 are different super instructions in the specification.

The Applicant has amended the specification as requested by the Examiner.

Rejections under 35 U.S.C. § 112

Claims 12-24 were rejected as being indefinite. Particularly, the Examiner stated that in claim 12 it is not clear what is a super instruction and how a super instruction is executed.

The Applicant has amended claim 12 to improve its clarity. In particular, claim 12 now recites that a super instruction comprises a linear block of code including instruction sequences to be executed by each of a plurality of processing sections and one or more branch instructions. Each processing section comprises a local memory for storing instruction sequences of the super instruction that are to be executed by that processing section. Each processing section also comprises a function unit for executing instructions stored in its local memory according to machine cycles, each function unit executing one instruction per machine cycle. Also, each processing section comprises a pointer containing a value defining the next instruction in said local memory to be executed by the function unit, wherein the pointers in each of the processing sections are reset to a new value determined by a target address of one of the branch instructions when a function unit branches in response to that branch instruction.

The Applicant submits that claim 12 is now sufficiently clear regarding what is a super instruction and how it is executed. In view of the above, the Applicant submits that claims 12-24 are not indefinite.

Rejections under 35 U.S.C. § 102

Claim 12 was rejected as being unpatentable in view of Wolf et al., "A variable instruction stream extension to the VLIW architecture" (hereinafter Wolf et al.).

As explained above, the Applicant has amended claim 12 to improve its clarity. The Applicant submits that amended claim 12 is allowable over Wolfe et al. In particular, claim 12 recites that the pointers in each of said processing sections are reset to a new value determined by a target address of one of said branch instructions when a function unit branches in response to that branch instruction. As explained in the Applicant's specification at page 8, lines 1-2, when one of the function units executes a branch, the branch target is broadcast to all of the function units. Each function unit receives an address which is loaded into its pointer register and which specifies a location at which the function unit is to resume processing. Applicant's specification at page 8, lines 6-8. Thus, when one function unit executes a branch, this controls all of the other function units to also branch without a branch condition having to be independently tested and executed within each of the other function units. As a result, programs can be simpler and more compact.

Wolf et al. does not disclose such a feature. Rather, Wolfe et al. describes a model architecture (referred to as "XIMD") that requires that each processor independently test its own condition for branching. This is clear because Wolf et al. explains at page 4, Section 1.4 that while each processor can provide information to other processors, "XMID uses fully distributed control" in which "no processor can control another processor." In addition, at page 8, Section 2.2, under the heading "Control Path," Wolfe et al. explain how each functional unit has its own control path by which it determines whether to branch based on its own testing of branch conditions.

The difference is that in claim 12, a branch occurring in one of the function units causes the other function units to be commanded to branch, whereas, Wolfe et al. teaches that each processor independently determines whether to branch based on its own testing of branch conditions.

The Applicant submits that amended claim 12 is allowable for at least this reason.

Conclusion

In view of the above, the Applicant submits that all of the pending claims are now allowable. Allowance at an early date would be greatly appreciated. Should any outstanding issues remain, the examiner is encouraged to contact the undersigned at (408) 293-9000 so that any such issues can be expeditiously resolved.

Respectfully Submitted,

Dated: Dec. 14, 2004

Derek J. Westberg (Reg. No. 40,872)